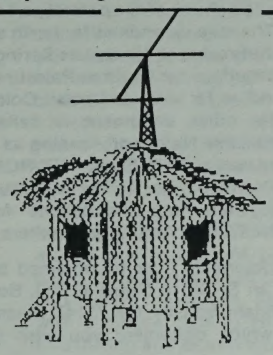
Digital Digest

Vol. 2 No. 1 *Devoted entirely to Digital Amateur Radio Communications* Jan/Feb 1989

In This Issue:

There's all this, and a whole lot more In this Issue!



From The Publisher's Shack

Happy new year everyone! This is an exciting year for us as it marks the official beginning of Digital Digest's first year of publication. For those of you who signed up as charter subscribers prior to publication of our premier issue, I thank you for your support. I hope that you enjoyed the premier issue and that you and new subscribers will find this and future issues even better.

A word about your subscriber mailing label. We have received several letters from you inquiring why the expiration numerals are 6-89 since you bought a one year subscription. The "6" is not the month of expiration, but the issue at which your subscription ends. The "89", of course, is the year.

All annual subscribers with subscriptions beginning with the Jan/Feb.'89 issue therefore have labels indicating 6-89. The "6" indicates the Nov/Dec.'89 issue. The only exception to this was our premier issue which was indicated as Vol.1, No.1. We are beginning this first full year of publication with Volume 2, Number 1. Since Digital Digest is published bimonthly, there are 6 issues per annum. The reason for this label system is so you will know at a glance with what issue your subscription will expire.

To keep cost of paperwork and postage down, we will not be sending renewal invoices or reminders. This is why it is important to keep an eye on your mailing label. To insure uninterupted service of your subscription, simply renew prior to publication of the next issue following your subscription's expiration (or earlier for that matter). I thank those of you who questioned our mailing system and appologize for the confusion. I hope this clarifies things.

As always, we encourage your input to the Digest, it is only in this way that we can mold this publication to best serve your interests and needs. We are commited to the Digest and hope that it will serve you well in the months (and years) to come.

We are excited to announce the debut of a new column, *The Programmer's Notebook* by Pete Smith, N4ZR. Pete is an experienced programmer who will be lending his expertise in the areas of writing software for Amateur Radio and reviews. Another seasoned pro, Stan Horzepa, WA1LOU, will also be contributing information on various digital topics.

With the new year, we all tend to make resolutions to improve ourselves in some way with a renewed commitment. This also could apply to improving our amateur numbers. Much has been said lately on the status of amateur radio and its lack of ability to transfuse new blood into our hobby (today's average age is 51). Several of our amateur magazines have recently editorialized on what has been done and/or what should be done to improve our aging numbers. Novice Enhancement was instituted along with strong appeals for a "no-code" operating license. Adding fuel to the "nocode" controversy are the reports that Morse will no longer be used aboard ocean going vessels, which are giving way to the new communications technologies. Novice Enhancement seems to have been less than successful in attracting new members. In fact, according to one recent W5YI Report, not only are there fewer novices now than two years ago, but fewer applicants became hams in 1988 (with Novice Enhancement),than in 1986.

Amateur Radio is a great hobby, perhaps the greatest, we all as active hams know that... so what gives... and why aren't we attracting the new blood our hobby so desperatly needs? There is obviously no easy answer, or solution, since greater minds than mine are pondering the same question. I can only offer my own perception of the problem which is this.

Like it or not, the face of amateur radio is changing. We're getting more wrinkles. We who gained access to this great hobby through the vacumm tube, and marveled at the advent of the transistor, and revelled at being able to home-brew our gear, are trying, and perhaps too hard to keep old traditions alive... and traditions die hard. Today's technology however, has made it virtually impossible to do any major building or servicing of our own gear.

(cont'd on page 2)

Digital Digest
The DIGIPEATER
News & Information
communicated from various sources.

(cont'd from front page)

And CW, for those of us who have enjoyed pounding brass, and still enjoy doing so, just maybe has been taken over by technology. We should not give up pounding brass if we enjoy it, but for the new breed of hams we are trying to attract, who are used to communicating via modem at baud rates of 2400 or so, CW may seem just a tad slow. It is becomming increasingly difficult to "sell" the need for CW proficiency as a prerequisite to obtaining a ham license.

The whole point of my editorializing here, is that for those of us "diehards", we just might have to accept a new computer based generation... computers are in the classrooms, office, home, trains, planes, boats, automobiles...and in ever increasing numbers in the ham shack. We're all learning and communicating by computer these days from anywhere to anywhere. What seems to me a natural means for gaining a new influx of amateur radio operators is to try tapping into the many "landline communicators" who are out there. Many who may not even know amateur radio exists. By doing so we could tap new resources of knowledge in software and hardware design which could help in better developing our amateur means of communications ... now so predicated on computer based technologies. Homebrew today might be redefined as creating new software and hardware for better packet operations and so on. We might access this new crop of hams by advertising and publishing articles in the varied computer hobbyist magazines, speaking in classrooms, talking with computer SIG's, etc., perhaps with an exclamation like "Communicate without wires all over the world... yes, it is possible with Amateur Radio!"

Tom Arvo, WA8DXD
Publisher

AMIGA RTTY PROGRAMS

Now here's an offer that's hard to beat. Ben, N4EJI, reports that he has terminal programs, as well as other programs that deal with Amateur Radio subjects. They are for the AMIGA computer and are available free of charge. You can find them on CompuServe (AMIGA Technical Sig), or you can catch Ben on the air, or drop him a line at his callbook address.

These programs cover SSTV, FAX (direct satellite FAX on 137 Mhz), RTTY and Packet Radio, and are specifically designed for the AMIGA computer. If hardware is required, then information is available on how to either build or acquire same.

-Source: RTTY Journal-

NET/ROM AIDS AVAILABLE

Doug Everitt, N5DUB, editor of "The Node" NET/ROM newsletter, is offering two documents that assist NET/ROM users and operators.

One document is the "N5DUB NET/ROM Map," which charts NET/ROM, TheNet, TexNet and other network devices located in a good chunk of the central part of the US. The map extends as far north as Omaha, Nebraska, as far east as Springfield, Missouri, as far south as Palestine, Texas, and as far west as Lamar, Colorado. The other document is called "Using 'Reliable Neighbor' Routing as a Friendly Alternative to Fixed NET/ROM Routing," which was written by Doug expressing his recommendations for optimal NET/ROM routing parameters.

These documents may be obtained by sending an SASE to Doug at P.O. Box 76452, Oklahoma City, OK 73147. Please specify which document you wish to receive.

-Source: Gateway-

Digital Communications with Amateur Radio...

is the title of a new book developed and published for AEA by Master Publishing.

If you haven't yet taken the plunge into the digital modes but are eager to learn more (which could be why you're reading D-D), this book is for you. The publication starts out with a brief introduction on the history of digital communications and rapidly moves on to the present technologies of computers and peripherals in the ham shack.

All of the amateur digital modes are covered including the means and methods of operation. Packet radio receives the most coverage including a chapter on Packet satellite communications.

By the way, if the AEA publication bears a strong resemblance to one distributed by Radio Shack, it's because the basic book was written by Jim Grubbs, K9EI and share the same publisher. In cooperation with the Publisher, the special AEA edition was adapted from Jim's book and modified to include AEA specific information with contribution's from AEA's Mike Forsyth, N7KQE.

While this book may not make you a digital expert overnight, it does offer a concise source of information from which to begin your ventures into the exciting world of digital communications. As we all know, there is no substitute for hands-on operating when it comes to learning, but "Digital Communications with Amateur Radio" might be just the thing to help get you started in the right direction.

More Antarctic Activity

Roman "Charlie" Cholawinsky, VK6MP/VK0MP, plans to conduct a digital DXpedition to Antarctica during 1989. The Australian Department of Transport and Communications has issued him a special call sign (VK0MP) for use while operating from Antarctica.

VK0MP will be stationed at Casey, one of the Australian National Antarctic Research Expedition (ANARE) bases as Communications Officer. For the past five years, Roman was in the Royal Australian Air Force working in the communications field.

The ANARE expedition is scheduled to last for approximately 13 months. During this period, VK0MP plans to use an AEA PK232 to operate in all Amateur Radio digital modes including packet radio.

The details of VK0MP's intended operating schedule are not yet final, however, a six-week survival training course for the expedition was scheduled to conclude in Tasmania near the end of November prior to arriving in Antarctica.

An impressive QSL card commemorating VK0MP's operation from Antarctica is available for contacts with the station and can be obtained from his QSL manager, Gil Mays, VK6AGC, at P.O. Box 53, Hillarys, WA 6025, Australia. Please include sufficient IRC's for return postage, otherwise, QSL's will go by way of the VK6 QSL bureau.

When available, regular bulletins detailing the VK0MP operating schedule and frequencies will be posted on PBBS's. from Gil Mays, VK6AGC

-Source: Gateway-

ARIES-1 Software Review

Now that the much awaited for RTTY Journal/CQ WW RTTY contest is behind me I am happy to report that some 300 plus QSO's later I am not too much worse the wear thanks in large part to a great software package for MS-DOS machines and PK-232 or KAM TU's called ARIES-1.

The ads found in current issues of CQ and 73 magazines tout the ability of the software too insert the date and time from the computer and the frequency and mode of the transceiver (if properly interfaced) into the log of the QSO. Also mentioned in the ad is the contest mode of logging, the automating dupe checking capability, the option to exit to DOS while ARIES remains resident in RAM and the support of all functions available through the terminal unit you are using.

(cont'd on page 3)

Aries-1 *(cont'd from page 2)*

Most importantly the software also supports a mouse for data entry, and herein lies the greatest work saving and nerve soothing concept currently available on the RTTY/PC software market.

Now for the specifics. ARIES-1 supports all modes of the PK232. The mode is changed quickly from the menus on the screen or with a simple mouse click. I have not done a lot of CW or AMTOR with the software but I did verify that they worked as claimed. The same is true with the Packet mode of the package which changes from HF to VHF mode with another mouse click or function call, it certainly worked well enough to let me make my nightly checkin's on the local BBS. ARIES-1 shines brightly on RTTY and works as good as any software I have used.

The outstanding feature of ARIES-1 is the real time logging program. All usual QSO info can easily be input from the screen. Once a call is entered into the logger, a search is automatically initiated and if a match is found, the information is put on the screen. This is quickly accomplished as the complete log is read into RAM every time the program is booted up. New log entries are immediately written into the log on command so that the most that could ever be lost is the latest contact (especially important in the contest mode).

Operating in contest mode was the real test. As I previously mentioned, the mouse is a great feature. Point the mouse to the ID field in the logging section of the screen and click after a target station's call prints on the screen, and presto, the call has been entered and a dupe search automatically started, all with a mouse click. Go back to the station with F1 (key transmitter) F4 call exchange (call of station clicked in plus de and your call), then one of the usual exchange buffers. Further data can be entered into the logger with mouse clicks in the same fashion. Point to RST and click the mouse after your report, point to state or name, or whatever - then click the mouse and it's on the screen. Finish the QSO with a mouse click to enter everything into the log and search for a new contact.

It is important to mention that the method of clicking the mouse at the right time takes a little learning. It must be done after a space and during the transmission of the word following the target word. This was simple to learn and made the entry of nine out of ten calls easy, but because of the many different ways that the rest of the contest exchanges were sent, it made entering other info (RST, zone, state, etc.) difficult. After a few tries, I abandoned this and kept track of that info separately, choosing to write it out rather than enter it with the keyboard. It is not necessary to use a mouse as all functions are supported without one, but it sure makes things a lot easier.

Also supported by ARIES are replaceable string parameters. This allows the embedding of special control characters into your stored buffers so that once the info is entered onto the logging screen, it can automatically be retransmitted as with the F4 call exchange function already mentioned. Included as a replaceable string parameter is the log entry number which would work as an automatically incrementing QSO number in a contest like BARTG.

This is the area I had planned to list the three or four faults that I had found with the ARIES-1 package, however, version 1.3 just arrived and two of those items are

(cont'd on page 17)

by Lynn W. Taylor, WB6UUT
463 Myrtle Street • Laguna Beach, CA 92651

★ Optimizing Packet Performance ★

Last issue we talked about getting started -- how to plug in your shiny new TNC, how to get your computer talking to it, and how to get things 'close enough' to make your first contact.

For most packet operators, this is as far as they go: obviously things are OK because they are able to talk to others.

WRONG.

This month, our topic is optimization: why to optimize and how to get the most out of your packet operation. Another side of this discussion is how to minimize the impact of your operation on the network and comply with the spirit 47 CFR 97.85.

Optimization falls into two categories, hardware and software. Hardware optimization deals with the radio and modem section of the TNC. Software optimization covers TNC parameters - mostly we'll be talking about avoiding collisions here.

Hardware optimization:

For hardware optimization, our primary goal is reducing the Bit Error Rate, or BER. The concept is simple: if a packet is received with an error, it is going to have to be retransmitted, if no error occurs, the receiver acknowledges the frame and no additional network time is used.

Let's assume for a moment that the average information frame contains 80 bytes or one line of text, and that we are connected directly. The total packet would contain 99 bytes, or (ignoring bit stuffing) 792 bits. If we have a bit-error-rate of 0.1% (1 in 1000), only 21% of our packets will be received correctly. If we can reduce the BER to 0.05% (1 in 2000), the odds improve significantly: 60% will be received without error, and with an error rate of 1 in 5000, 99.8% of all packets sent will be received correctly.

There are two simple adjustments which any amateur can use to reduce the BER.

First, check the adjustments in your modem: most TNCs use the EXAR 2206/2211. The 2206 modulator is adjusted to give the correct tones (for VHF, 1200 and 2200 Hz) and the 2211 demodulator is adjusted for a center frequency of 1685 Hz. My experience has been that these require adjustment after the first month of operation, after 3 months, after 6 months, after the first year, and approximately annually after that. With the built-in calibration routines in practically all TNCs, there is no reason for these tones to be off.

The second parameter is audio levels. If at all possible, try to arrange your radio and TNC so your audio levels will be constant: microphone inputs with front panel adjustments and speaker outputs with adjustable volume should at least be marked with the correct settings if you can't simply dedicate a radio to packet.

Optimize the receive level first: find a busy channel (not difficult these days), set your TNC to the highest monitor setting (typically MON 6 or MON Y), and adjust the receive volume until you successfully receive packets from the widest variety of stations. This should be your optimum receive level.

For transmit audio, monitor your own signal in another receiver and, using the CALIBRATE command in the TNC key the transmitter and select the higher of the two tones, then increase the audio until it quits getting louder, then lower it until it just starts to go down. This will get you close.

Next, pick a station nearby, and connect to yourself by digipeating through it. Send a few packets and find the lowest audio level that works **with that station,** then find the highest level. Somewhere between these two levels will be an optimum level for reception by that station.

Repeat this procedure with several stations, and note the optimum level for each one. The optimum transmit level for stations in your area will be the average of these levels.

If you like, you may also adjust the receive audio while connected to yourself through several stations and optimize it that way.

These adjustments alone will give you an edge on your local network.

Software Optimization:

The next area we can optimize is software. Our main goal here is to reduce the number of collisions, maximizing throughput.

Before I go further, let me define throughput: Throughput is the total number of bytes of data which are successfully received and acknowledged on a given network during a given interval of time.

In one direction, longer packets increase throughput by reducing overhead (reducing the number of headers).

In the other direction, longer packets are more likely to contain errors, and are bigger 'targets' for noise and collisions.

In other words, packets should be as long as possible, and no longer.

Let's take the example from our discussion of Bit Error Rates: with a BER of 0.1%, our 792 bit packet would be received correctly 21% of the tjme. Cutting the packets to 392 bits (setting PACLEN to 30) would allow packets to get through 61% of the time. For the 1 in 5000 case, stretching packets to 1432 bits (PACLEN 160) would still get through 71% of the time without error.

Our first moral: set PACLEN appropriately for the conditions in effect -- short packets for poor conditions, longer packets for file transfers when things are good.

The next area to optimize is timing. As an example, two stations, (Rick and Steve) get two brand new TNCs. They take 'em out of the box, and put them on the air using the default parameters.

Steve and Rick are connected to stations through a digipeater. Steve types, then hits return at the end of the line. Because the digipeater is transmitting right now, Steve's TNC waits for the channel to clear.

Rick starts typing about the same time Steve does, and hits return at the end of a line. Since Rick's TNC also hears the digipeater, his TNC waits for it to finish.

When the digipeater stops transmitting, both Rick and Steve wait the normal DWAIT period, then start transmitting at **the exact same time!** We all know this as a collision.

Since neither station captured the digipeater, neither was repeated, and neither Rick nor Steve get an acknowledge for their packets.

Since the packet was not acknowledged, Steve's TNC waits 9 seconds (the standard interval for FRACK is 3, and it is multiplied by 3 to allow for the digipeater). Rick's TNC also waits 9 seconds. Then, both wait for a clear channel and transmit again, **at the exact same time!**

As you can see, unless something else (a user that Rick hears, but Steve doesn't, for example) knocks them out of sync, they are liable to collide almost forever. Other random factors in the TNCs help break the deadlock, but there is a better way.

If Rick changes his setting of FRACK to 5, then, when the first collision occurs, Steve waits 9 seconds and grabs the digipeater, and Rick waits 15 seconds and grabs the digipeater -- 6 seconds after Steve -- with no collision.

Our second moral: raise FRACK to let others go ahead when a collision occurs. Personally, I usually have FRACK set to 7, and under poor conditions will raise it to 11 or 13. Also, if you are digipeating to talk to another station, ask what his FRACK setting is, and set yours to some other value.

On a channel with mixed BBS operations and keyboard-to-keyboard operations, the Bulletin Board Systems should run higher values for FRACK (and DWAIT) to give priority to the QSOs in progress. Since QSO traffic sends packets seldom, the slower timing will be offset by fewer collisions (and retries).

Another parameter which can be adjusted is TXDELAY. This is the time allowed for the transmitter to key and for the receiving station to detect the carrier and 'sync' to the incoming signal. The default is 300 milliseconds, or 3/10ths of a second. Our 792 bit packet only takes about 6/10ths of a second to send, so half of the time is spent waiting for things to get started. Next time you operate, try adjusting TXD lower until retries start to increase, then raise it just above the minimum. Remember, setting TXDELAY too low will prevent some stations (with slow receivers) from connecting to you. Setting it to match the fastest guy around will only allow you to talk to that one station!

p-Persistence, or the joys of relaxing:

Not all TNCs have variable persistence, but if yours does, by all means use it.

Let's visit our friends Steve and Rick again. Part of their problem comes from the 1persistent nature of their TNCs -- if the TNC sees a free channel it goes for it.

Jeff and Greg also have new TNCs, and they both have pPersistent capabilities. Geographically, there situation is identical to Steve and Rick: both see a common digipeater, but they don't see each other. They set their TNCs to be 0.5- Persistent (the value for p is 0.5 instead of 1).

Both Jeff and Greg type a line and hit return while the digipeater is transmitting, but, when the digipeater unkeys, Jeff's TNC flips a coin, and, if it's heads, he transmits, tails he doesn't. Greg's TNC also flips a coin, and decides to transmit based on the toss.

So, at any given transmitting opportunity, one of four things is bound to happen:

		Greg: Heads	Greg: Tails
Jeff	Heads	Both	Jeff
	Tails	Greg	Neither

Of the three cases where stations transmit, a collision is avoided two out of three times!

The accompanying program *(below)* and table *(pg. 15)* show the effect of various values for p when different number of stations wish to transmit. The table is based on the assumption that the number of stations on the vertical side all have traffic to offer at the same time. The entries in the middle of the table are the number of collisions which occur under these conditions.

Our last "software moral" is to relax. Remember, this is a hobby anyway, it doesn't really matter if it takes a few seconds longer for a line of text to get through. If you try 'backing off' you'll probably find things going faster, especially on busy channels.

Optimal Operating Practices:

Finally, there are a couple of things you can do when you operate to help the network:

If there is a layer three network in your area (NET/ROM, ROSE or TEXNET), use the network facilities instead of digipeating if a digipeated frame gets lost, it has to go through every digipeater again, while the smarter networks can fix the problem where it happens.

If you are transferring files or reading lots of messages from a Bulletin Board, avoid 'prime time' and respect the operating habits of others.

'DXing' bulletin boards is generally a poor practice, but if you really have to, try to limit this activity to off-peak times.

Next issue we'll embark on a discussion of protocols, starting with an in-depth look at AX.25 and proceeding from there into a more general discussion of the ISO Open Systems Interconnect model, contention and noncontention based networks and other fun stuff.

```
Program Simulator;

uses CRT;

const p:array[1..4] of real=(0.5, 0.25, 0.1, 0.01);

var NumberOfStations, StationCtr, Tries, Transmitters,
    Collisions, pCounter: integer;

Function Station(p: real): boolean;

{ function provides a crude simulation of what happens at a }
{ station when the channel is clear for some value of P.    }

Begin
 Station:=Random(256)<(p*256)
End;

Begin
  ClrScr;
  For pCounter:=1 to 4 do
    Begin
      GotoXY(pCounter*10, 2);
      Write(p[pCounter]:1:2)
    End;
  For NumberOfStations:=2 to 10 do
    Begin
      GotoXY(1, NumberOfStations+2);
      Write(NumberOfStations:2)
    End;
  For pCounter:=1 to 4 do    {for each value for p in the table}
    For NumberOfStations:=2 to 10 do {and anywhere from 2 to 10
                                      stations}
      Begin
        Collisions:=0;
        For Tries:=1 to 10000 do  {simulate 10,000 transmitting
                                   opportunities}
          Begin
            Transmitters:=0;
            For StationCtr:=1 to NumberOfStations do
              If Station(p[pCounter]) then Inc(Transmitters);
            If Transmitters>1 then Inc(Collisions);
          End;
        GotoXY(pCounter*10, NumberOfStations+2);
        Write(Collisions/100.0:4:1,'%')
      End
End.
```

by Norman Sternberg, W2JUP
279 Adirondack Drive • Farmingville, NY 11738

Packet NTS Operation With A PBBS

An unofficial and brief tutorial

(Except where indicated otherwise, the comments and recommendations contained in this tutorial are solely the opinions and viewpoints of the author. These remarks are based on several decades of experience in commercial, MARS and amateur traffic handling, combined with four years of 24-hour daily operation of a busy packet BBS and message switch. The author does not represent these remarks as approved by the NTS, the ARRL, or any other organization or group. Readers should not infer such approval or sanction by the NTS, its officials or the ARRL. Always check procedures with your local NTS people or the ARRL Section Traffic Manager.

PREFACE

One thing must be recognized before we start. Packeteers **must** understand and respect existing NTS operations developed over the years on modes other than packet. NTS operations and organizations have been around on CW, voice and RTTY modes long before packet radio loomed on the NTS horizons. Packeteers **must** consider the philosophies and experiences gained by NTS peope during many years of traffic handling. Packeteers **must** respect existing and traditional NTS operating practices, even if those practices may appear at times to be awkward, cumbersome or possibly less than 100% efficient in packet radio.

Packet NTS operations and operators MUST adapt to established NTS methods and NOT try to "revolutionize" NTS operations without the consent and agreement of the existing NTS community. NTS principles must be the overriding consideration, regardless of possible advantages shown by packet's unique operational procedures and methods.

With this in mind, before we get into the specifics of NTS message filing and retrieval in Packet BBSs, let's start at the beginning to ensure that we have a clear picture of how the NTS messages are supposed to be drawn and handled in general, and without specific regard to packet radio.

1: USE THE STANDARD AMATEUR MESSAGE FORM

As a starting point, the following information was compiled from ARRL form CD 218 2/83 by the NLI Section Traffic Manager, Bob Smutny K2MT, and is shown here in its original form and content.

AMATEUR MESSAGE FORM

Please see also the sample message shown on page 18.

Every message should have the following information in the order given:

(1) Preamble

(a) Message number (starting from 0 each month or year)
(b) Precedence (R,W,P or EMERGENCY),(See below)
(c) Handling instructions (Optional),(See below)
(d) Station of origin (Call who first sent message)
(e) Check (Number of words/groups in text only)
(f) Place of origin (Town where message came from)
(g) Time filed (Optional),(Time when first sent)
(h) Date (When first sent)

(2) Address

(a) Full name (First and last)
(b) Full street address (Include apartment number)
(c) Town name, state and zip code
(d) Telephone number (With area code)

(3) Text (Actual message),*(Try to limit this to 25 words or less)*

(4) Signature

(5) Routing record

(a) Call of station you from whom your received the message
(b) Call of station, or person, to whom you sent or delivered the message

RULE 2: REMEMBER THAT THE PACKET BBS MAY BE UNATTENDED

Many (if not most) packet BBSs are unattended for many hours and even days at a time. Unless NTS traffic is appropriately addressed **when you send it to the PBBS,** your outbound NTS message may possibly **sit and rot** on that PBBS until the System Operator (SysOp) does his/her routine BBS "housekeeping" chores.

When (and if) the SysOp notices the traffic sitting there, going nowhere, the conscientious SysOp will take whatever steps are available to fix defective addresses and speed the traffic on its way. If the SysOp can't figure it out, most of them will send a SERVICE message back to the originator asking for clarification. This results in additional, needless delays. Delays caused by poor packet message addresssing give a bad name to both the NTS and the packet community and are a disservice to the fraternity as a whole.

With that clearly stated, let's get to the business at hand.

RULE 3: OBSERVE MESSAGE LENGTH RESTRICTIONS

As a general rule, the NTS community prefers (and in some areas insists) that NTS traffic be limited to 25 words. Where possible, packeteers accepting or filing (entering) NTS traffic for handling via packet channels should comply with these restrictions to the best of their ability. Packeteers must bear in mind the possiblity that a given message may have to be handled at some point in Morse or voice or RTTY.

RULE 4: FILE YOUR MESSAGE USING THE "ST" COMMAND

All NTS traffic **must be sent or filed** into the initial PBBS using the "ST" command. The BBS programs "know" the "ST" command as NTS traffic origination and set certain program flags to provide special handling to that message from that point on. Each BBS receiving and handling that "ST" message will convey similar special handling.

RULE 5: USE ACCEPTED STANDARD PACKET ADDRESSING METHODS

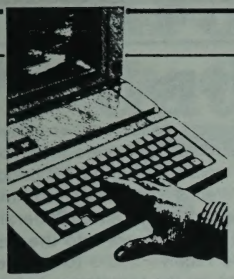
There are **three** generally accepted regional routing address methods.

1. NTS State Addresses - The letters "NTSXX", where "XX" is the Postal Service symbol for destination STATE

   NTS SECTION Addresses - The letters "NTSXX" or "NTSXXX" where "XX" and "XXX" are the ARRL Section abbreviations

2. ZIP Code Addresses — The five-digit Postal Service ZIP Code

(cont'd on page 17)



Toasters on the air...

Why did the moron throw his Macintosh out the window? He wanted to get his computer on the air!

Back in ancient times, when the IBM PC was king and Apple introduced the Macintosh computer, many computer gurus considered the "Mac" a toy or, as one sage put it, "a toaster that spits out pictures."

Times have changed. The 1989 flavor of the Mac is much more powerful than the original Apple offering and, as a result, the Mac has found new flavor in the business world and is coming into its own in the Amateur Radio World. More and more hams, even some former died-in-the-wool IBM PC-types, have seen the MacLight and have added a Mac to their shack.

Like the PC, the Mac is a powerful computer, however, unlike the PC, it is easy to harness the Mac's power. The Mac operating system is easy to learn and use. Most of the programs written for the Mac stick to the standard Mac interface. What this means is once you learn how to use one Mac program, you can usually load and run a new Mac program and immediately begin to use it without consulting the program's documentation.

For example, let's take two different Mac applications: SuperPaint, a graphics program, and Reflex, a data base program. If you want to use the print function in either program, you pull down the "File" menu, which is located in the same place in both programs, and select the "Print..." command, which is located in the same general location as the file menu. After selecting the Print... command, both programs present the user with similar "dialog boxes," which ask the user the quality of the printing he desires, the number of copies he wants printed, etc. Other common commands function the same way in all Mac programs that follow the standard interface. Therefore, you do not have to relearn every function each time you use a new program.

Besides the simplicity of using the Mac, Mac is also noted for its graphic capabilities. Just look at the accompanying figure that was generated by DX Window for an example of the kind of pictures the Mac can spit out. There is no doubt about what the figure represents. The globe looks like the one we all know and love as depicted by the ARRL Amateur Radio Map of the World. It sure does not look like it was put together using the toy blocks that some computers use for graphics.

Another feature of the Mac is its EIA RS-422 serial ports which provide high-speed transfer for information between the computer and peripheral devices such as ham radio interfaces, TNC's, modems, etc. I have used various TNCs connected to my Mac with the serial data rate set at 9600 baud. Received packets seem to pop onto the Mac's display a whole packet at a time... it gives you a real feel for the speed of packet radio data communications.

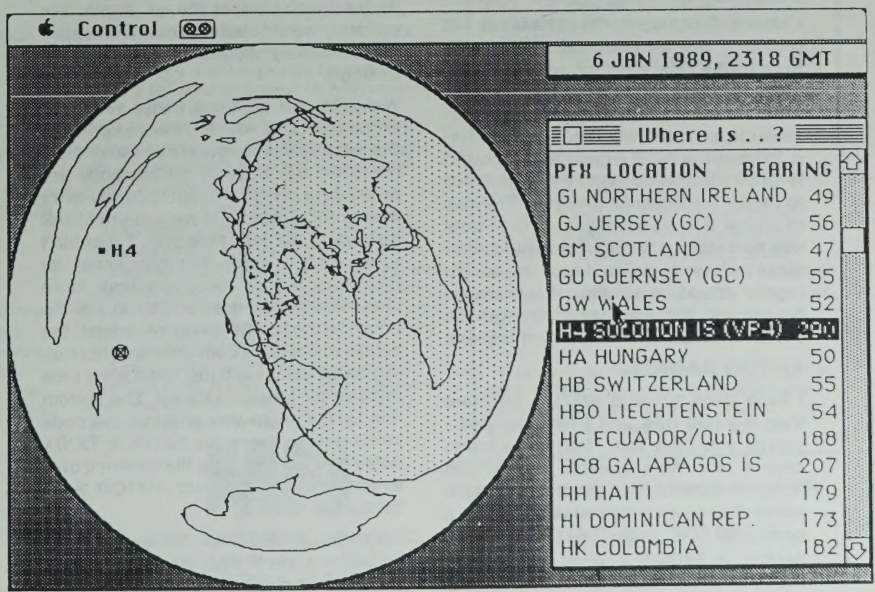
Each month there are more Mac applications available and more are on the way. And, there is always Mac BASIC, Pascal, C, etc. for those who are willing to convert other computer Amateur Radio programs over to the Mac.

The Mac's future in Amateur Radio is promising. Standby and see what else pops out of the Apple toaster.

by Stan Horzepa, WA1LOU
75 Kreger Dr., Wolcott, CT 06716-2702
CompuServe' 70645,247

DX WINDOW
is an Amateur Radio application for the Macintosh computer. It builds an equidistant azimuthal map of the world centered on your location and displays the gray line for any desired time. Select a DX location in the "Where is...?" menu and the antenna bearing to that location is displayed and the location is indicated by a black block on the map. DX WINDOW is available from:

Engineering Systems, Inc.,
P.O. Box 939,
Vienna, VA 22180



by Paul Newland, AD7I

Post Office Box 205 • Holmdel, NJ 07733-0205

★★★ Timing & Other Considerations ★★★

Greetings! Last issue we talked about the APLink system that provided an AMTOR BBS using a PC and an AMT-1 (or PK-232). Well, since that article I have sent out many copies of the code to people all around the world. It will be interesting to see how many stations we have on the air in the next few months. Again, if you would like a copy of the APLink code, send me a formatted 5-1/4 inch 360K floppy and a self-addressed mailer with enough postage stamps to get it back to you.

A NEW GADGET

Although it doesn't use AMTOR, if you like gadgets that make use of low-speed "data communications", you might want to take a look at the November issue of the ARRL's QEX magazine, starting on page 8. There Karl Williamson, KA0BUA, outlines an Emergency Broadcast System (EBS) decoder. That's the system that is designed to notify the public of impending danger (from "shoot the nukes" to more mundane things like flood warnings) by way of commercial broadcast stations. The alert system is simple: when the AM/FM/TV station wants to alert the public they send a 20 second blast of two tones (960 Hz and 853HZ), simultaneously. When the decoder detects these signals it sets a latch and connects the speaker to the radio's audio output. The listener can then hear the voice message that follows. Normally, the decoder disconnects the speaker from the radio so the user doesn't need to hear the regular broadcast program.

I plan to build up Karl's system but will probably use an inexpensive AM radio from Radio Shack for the receiver (I got mine on sale for about $7) although Karl's article outlined how to make an AM receiver from scratch. One addition I will do is to snag the AGC signal from the radio to drive a signal strength meter and also to detect the absence of signal. The loss of signal will sound a buzzer to tell me that somethings wrong. Because I live so close to New York City, I have the advantage of choosing a station that is always on the air. If I lose signal strength it's probably because my radio went haywire!

AMTOR RADIOS

I have been running AMTOR for more than 5 years now and I have noticed... maybe you have too... that some transceivers are more suitable for AMTOR communications that others. Let's take some time to explore just what makes a good AMTOR radio (all these items, except for switching times, also apply to plain old RTTY as well).

The first requirement that everyone should be familiar with is switching time. That's the time it takes the radio to go from transmit to receive (TX/RX) or receive to transmit (RX/TX). A year or so ago I wrote an article for QEX that outlined a measurement procedure to test the switching times of radios. In that article I defined the RX/TX time as being the time to go from receiving to transmitting a signal at the proper frequency with at least 25% (-6 dB) of the final full output power. The TX/RX time was defined as the time to go from transmitting full power to receiving with the recovered audio voltage of more than 50% (-6 dB) of its final value. These values, although arbitrary, provide a convenient reference point for making measurements. Also, they are points at which useful data can be transfered. To wait until the voltage reaches 90% of maximum (like the text books tell you to do for measuring rise-time) is foolish. Your modem will probably work fine when it has only half the usual audio voltage at its input -- no need to wait for 90% of maximum voltage.

Switching times are important in AMTOR when using ARQ (chirp-chirp) communications. If the RX/TX switching time is too slow the AMTOR code converter begins to send data before the radio transmitter is up to power, causing the first bits to be transmitted at reduced or even zero output power. This certainly would lead to errors and you probably couldn't get any data through. If the TX/RX switching time is too slow the receiver may miss the first few bits that it should hear just after its transmitter leaves the air. Again, this certainly would lead to errors and, again, you probably couldn't get any data through.

Well, OK, we all know about switching times now and why they are important. But what are some good numbers? Well, times of less than 20 milliseconds are pretty darn good. I wouldn't pay any extra money for a radio that switches in 15 mS instead of 20 mS. However, I wouldn't mess with a radio that had either an RX/TX or TX/RX switching time more than 40 mS. Yes, it will work most of the time, but you will need to adjust the delays in the code converter and increasing those delays will just reduce the time allowed for blocks in transit. The bottom line is that when you program the code converter for excessive RX/TX or TX/RX delay you are reducing the distance over which you can conduct AMTOR ARQ contacts.

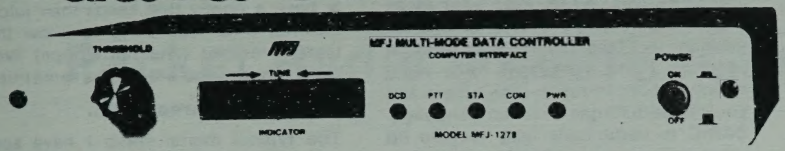
Another consideration for the AMTOR operator is the IF filter bandwidth of the receiver. On some of the modern amateur rigs the designers assumed (wrongly!) that if you selected SSB mode then you must always want to use the SSB bandwidth filter. Well, for AMTOR, nothing could be more incorrect. I always operate AFSK. I plug the TX tones into the mic and route the receive audio to the demodulator and set the selector switch to SSB. The problem here is that many of the "hightech" rigs force you to use the filter for the mode you have selected. Thus, for AFSK you are often stuck with the SSB filter. Well, for AMTOR, you really want that 300 or 500 Hz filter in the IF... not the 2.4 kHz SSB filter. The Drake 4 and 7 line always assumed the operator had a head on his or her shoulders and that it could probably be used to good advantage. The designers at Drake demonstrated their confidence in the user by permitting filter selection independent of mode. Collins designers did the same. Oh what a joy it is to run AMTOR with a narrow filter! Some people with the whiz bang radios have done some mods to allow them to switch in the CW filter when operating AMTOR. At the risk of generalizing, it's probably worth whatever effort is required to get the CW filter selected when running AMTOR.

As a side note about proper IF filter bandwidth, when I got my first "superduper" rig, after finally making general (superduper for me then meant the rig could do phone and included a VFO), I found I still didn't like CW. That confirmed my feelings at the time that once one got on phone why would anyone want to use CW. Well... my problem was that my "super-duper" rig didn't include a CW filter. Eventually I traded up to a rig that had a CW filter and lo-and-behold, CW became fun. I found I could actually adjust the radio to hear only one station at a time rather than three at once. What a pleasure! Except for a couple of occasions, I haven't had a microphone connected to my HF rig for more than 4 years (no, I didn't discover the pleasures of CW just 4 years ago -- I just no longer have the urge to work SSB). If you are tired of hearing more than one AMTOR or RTTY station at one time consider adding a narrow IF filter to your rig. When restricting yourself to good modems, that $50 to $100 IF filter is a much better investment than buying a modem that's $50 to $100 more expensive than another.

Another issue, somewhat related to the IF filter bandwidth, is the audio tone frequencies that your demodulator wants. You need to ensure that the tones your radio outputs when the narrow IF filter is

(cont'd on page 18)

MFJ multi-mode data controller



9 modes for only . . . $249.95

Amateur radio's most versatile multi-mode data controller -- the MFJ-1278 -- lets you join the fun on Packet, AMTOR, RTTY, ASCII, CW, Weather FAX, SSTV, Navtex and gives you a full featured Contest Memory Keyer mode . . . you get 9 modes . . . for an affordable $249.95.

Plus you get MFJ's new **Easy Mail**™ so you and your ham buddies can leave messages for each other 24 hours a day.

You'll find it **the most user friendly of all multi-modes.** It's menu driven for ease of use **and** command driven for speed.

A high resolution 20 LED tuning indicator lets you *tune in signals **fast** in any mode.* All you have to do is to center a single LED and *you're precisely tuned in to within 10 Hz* -- and it shows you which way to tune!

Plus you get 32K RAM, KISS for TCP/IP, high performance HF/VHF/CW modems, software selectable dual radio ports, AC power supply and more.

All you need to join the fun is an MFJ-1278, your rig and any computer with a serial port and terminal program.

You can use the MFJ Starter Pack to get on the air instantly. It includes computer interfacing cable, terminal software and friendly instructions . . . everything you need to get on the air fast. Order MFJ-1282 (disk)/MFJ-1283 (tape) for the C-64/128 and VIC-20; MFJ-1287 for Macintosh; MFJ-1284 for the IBM or compatible. $19.95 each.

Packet

MFJ's new generation packet mode gives you genuine TAPR software and hardware plus many MFJ enhancements like Easy Mail™.

A new Kiss interface makes the MFJ-1278 TCP/IP compatible.

Extensive tests published in **Packet Radio Magazine** ("HF Modem Performance Comparisons")prove the TAPR designed modem in the MFJ-1278 gives better copy with proper DCD operation under all tested conditions than the other modems tested.

New AMTOR mode!

Now the MFJ-1278 has a new AMTOR and Navtex mode, making it the only controller to feature **nine** digital modes.

MFJ-1278 transmits and receives AMTOR and includes all AMTOR modes: ARQ (Mode A), FEC and MODE S (Mode B).

Baudot RTTY

You can copy all shifts and all standard speeds including 170, 425 and 800 Hz shifts and speeds from 45 to 300 baud. *You can copy not only amateur RTTY but also press, weather and other exciting traffic.*

You can transmit both narrow and wide shifts. The wide shift is a standard 850 Hz shift with mark/space tones of 2125/2975 Hz. This lets you operate MARS and standard VHF FM RTTY.

ASCII

You can transmit and receive 7 bit ASCII using the same shifts and speeds as in the RTTY mode.

CW

You get a Super Morse Keyboard mode that lets you send and receive CW effortlessly, including all prosigns -- it's tailor-made for traffic handlers.

A huge type ahead buffer lets you send smooth CW even if you "hunt and peck".

You could store entire QSOs in the message memories, if you wanted to! You can link and repeat any messages for automatic CQs and beaconing. Memories also work in RTTY and ASCII modes.

***A** tone Modulated CW mode turns your VHF FM rig into a CW transceiver* for a new fun mode. It's perfect for transmitting code practice over VHF FM.

An AFSK CW mode lets you ID in CW.

You also get a random code generator that'll help you copy CW faster.

Weather FAX

***You'll** be fascinated as you watch WEFAX signals blossom into full fledged weather maps on your Epson or IBM graphics compatible printer.

Automatic sync and stop lets you set it and leave it for no hassle printing.

You can save FAX pictures and WEFAX maps to disk if your terminal program lets you save ASCII files to disk.

Pictures and maps can be saved to disk or *printed to screen in real time or from disk if you have an IBM or Macintosh* with the MFJ Starter Pack.

You can transmit FAX pictures right off disk and have fun exchanging and collecting them.

Slow Scan TV

The MFJ-1278 introduces you to the exciting world of slow scan TV.

You can print slow scan TV pictures on any IBM or Epson graphics compatible printer. If you have an IBM or Macintosh you can print to screen and save to disk with the MFJ Starter Pack.

You can transmit slow scan pictures right off disk. If your terminal program lets you save ASCII files you can save pictures from over-the-air QSOs.

You can transmit and receive 8.5, 12, 17, 24, and 36 second black and white format SSTV pictures using two levels.

Contest Memory Keyer

***Nothing** beats the quick response of a memory keyer during a heated contest.*

You'll score valuable contest points by completing QSOs so fast you'll leave your competition behind. And you can snag rare DX by slipping in so quickly you'll catch everyone by surprise.

Message memories let you store contest call, name, QTH, rig info -- everything you used to repeat over and over.

You get iambic operation, automatic incrementing serial numbering, weight control to penetrate QRM and more.

More Features

Turn on your MFJ-1278 and it sets itself to match your computer baud rate. Select your operating mode and the correct modem is automatically selected.

***Plus** . . .* printing in all modes, threshold control for varying band conditions, tune-up command, lithium battery backup, RS-232 and TTL level serial ports, watch dog timer, FSK and AFSK outputs, output level control, speaker jack, key paddle jack, test and calibration software, Z-80 at 4.9 MHz, 32K EPROM, and socketed ICs. FCC approved. 9x1½x9½ in. 12 VDC or 110 VAC.

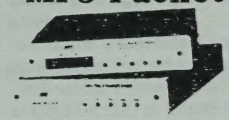
Get yours today and join the fun crowd!

New Firmware Update

A new KISS/AMTOR/Navtex Firmware update is available to MFJ-1278 owners.

MFJ's powerful update is the most reasonably priced mulit-mode upgrade by any manufacturer. Contact your dealer or MFJ for yours today!

MFJ Packet Radio



MFJ-1274 **$139.95**

MFJ-1270B **$119.95**

MFJ-1270B super clone of TAPR's TNC-2 give you more features than *any* other packet controller -- for $119.95.

You can double your fun by operating both VHF *and* HF packet because you get **high performance** switchable VHF/HF modems.

You get MFJ's new **Easy Mail**™ with soft-partitioned memory so you and your friends can leave messages for each other 24 hours a day.

In MFJ's new WeFAX mode you can print full fledged weather maps to screen or printer and save to disk using an IBM compatible or Macintosh computer with an MFJ Starter Pack.

A new **KISS** interface lets you run **TCP/IP.** They also come **NET ROM** compatible -- ***no modification needed!***

You also get 32K RAM, a full one-year unconditional guarantee and you can use 12 VDC or the included 110 VAC power supply.

For dependable HF packet tuning, the **MFJ-1274** gives you a high resolution tuning indicator that's accurate to within 10 Hz -- and it's only $20.00 more.

MFJ

MFJ ENTERPRISES, INC.
Box 494, Miss. State, MS 39762
601-323-5869 **Telex:** 53-4590 MFJSTKV

FOR YOUR NEAREST DEALER
or to order call toll free
800-647-1800
One Year Unconditional Guarantee

MFJ . . . making quality affordable

BITS & BYTES

by Lacy McCall, AC4X
3001 Zelda Drive • Montgomery, AL 36106

This column is a catch all of various topics and a few items that I find interesting. I am trying to avoid the temptation to get into a lot of detail, leaving that to others who will address the various digital modes; so don't fuss if I state something that you think not fully explained. Anyway it really doesn't matter. A few of the items on my mind currently, are included for what they are worth.

So You Want To Be A Programmer:

Every ham who operates in the digital mode has some experience with computers. Once into computers, most people become fascinated with the equipment and language possibilities and spend countless hours trying to make their computer do some special something... that's programming, and whether anyone who uses a computer will admit it, programming is an intriguing hobby by itself. Sometimes, the digital modes your computer manages become secondary to the desire to create a masterful computer program. Since, all computer nuts (and particularly Amateur Radio Computer Nuts), tend to charge off wildly when beginning a program, I would like to offer a few hints to take away the pain when you later find it time to de-bug!

Mr. Webster defines a BIT, as it relates to computers, as "a single digit in a binary number system...", and goes on to say "a unit of information equal to the amount of information obtained by learning which of two equally likely events occurred." What this says (I think) is that a ZERO is OFF and a ONE is ON, and there are no more options. That's the basis for your whole computer system, so never loose sight of that fact. The whole of programming is to test the OFF and ON and derive some logic from the test result.

The first decision you must make is what "language" to use. Beginners almost always start with BASIC, since it is most like english syntax and easily understood. If you really into programming and want to expand your computer vocabulary, there are multitudes of books available that discuss not only the various language details, but give "how-to" examples to get you started.

No matter what language you choose, many pitfalls are common. First, it has been my experience that it is easier to write a good program from scratch, than it is to modify a good program that may be similar, but is not tailored to your application. That includes programs of some length written earlier by you. There is nothing wrong with using other programs as a library source, but it is very easy to get completely entangled when modifying long programs, even when they are well documented.

So you should start from the top. Take a few minutes to organize your thoughts and objectives. Write down your ideas and take some time to arrange and re-arrange the task outline you want to program. Some languages force more structured organization than others, but you can be disorganized in any language. BASIC is particularly easy to gum up when you can put any number of GOTO's at any place in the program. Limit the routines you design to as few entry points as possible. This is not too difficult to accomplish, if you start with a plan.

The other thing I have experienced is that the way to learn a new computer language is to write a long program using the language. Currently, I spend most of my programming time with Turbo Prolog and 'C'. The languages are quite different in that Prolog is object oriented and depends to a large degree on recursion, whereas 'C' is structured and is based on function definition at it's lowest level. When I move from one to the other, it takes a complete re-thinking of the computer programming logic, but I learned both languages by writing terminal programs for PC Compatible computers.

The final tip to the new programmer is not to forget good reference materials. Not only can you get good books about the software and language applications, but a number of books such as Peter Norton's series are invaluable, when you start trying to control the hardware. Finally, don't be discouraged, take time to educate yourself so that you have at least a rudimentary understanding of your machine and the processor... and don't forget that a ZERO is OFF and a ONE is ON, and there are no more options.

Have You Noticed...Hams Live Long Lives...

Not long ago, I remember seeing an article in some ham magazine about the hazards of RF in the ham shack. At the time, I wondered how many hams had actually been affected by all that RF bouncing around, and I made a pledge to live longer and stay out of the ham shack. Well, that didn't last long and was soon forgotten. Then, a friend started talking about retirement. This particular friend had few hobbies and had spent most of his life performing the normal adult routines of working and raising a family.

Our discussion was about the fact that often you hear about people retiring, and soon after, becoming silent keys. Not really silent keys of the type we know, because most of the retirees that kick out early don't have anything to do; which brings me back to RF in the ham shack. I don't believe that it harms at all. At many a hamfest, I have noticed that are multitudes of old hams. They obviously have survived the RF beautifully, and obviously have a hobby that keeps their juices flowing. It is reassuring to know that hams live long (and full I hope) lives. Look around, you will see the same thing.

Operate A Stereo Digi...

The typical digital setup I have seen consists of a computer, a controller (TNC or equivalent), and rig. All the information to be sent is entered at the keyboard or from a pre-typed buffer and information received is printed on a terminal screen. In the old RTTY days where teleprinters were used, the clack-clack of the printer made the ham shack a noisy place; but not now. You can use a printer, but most of the time the loudest thing in the room is the fan keeping the computer cool. Now I know that lots of digital operators also monitor VHF and HF frequencies that do make a lot of noise on occasion, but what I recommend is that you try my technique.

Here's the outline:

1. Start your system and put all components on automatic.
2. Lean back in you chair. (which is best if it's comfortable).
3. Pour yourself some light refreshment.
4. Tune in your stereo to some nice quiet music.
5. Relax with a digital mode and monitor the world.

All this in living stereo. Digital modes are the only modes that allow the ham to avoid the noise created by CW and SSB, so go out today and buy a good stereo system for your hamshack. I'm sure your wife would prefer the Lionel Riche to the CW !

Let's Start A Controversy

Here's what I think:
Host Mode is redundant, unnecessary and at best cumbersome. That should get you going, particularly if you are one of those programmers (like me), who have written software that uses the Host Mode.

Most of the staff programmers and consultants that work for the manufacturers of TNC's spend hundreds of hours coding routines that are solid, properly debugged and well tested. The result is the ROM software that your computer terminal program can communicate with, easily and reliably.

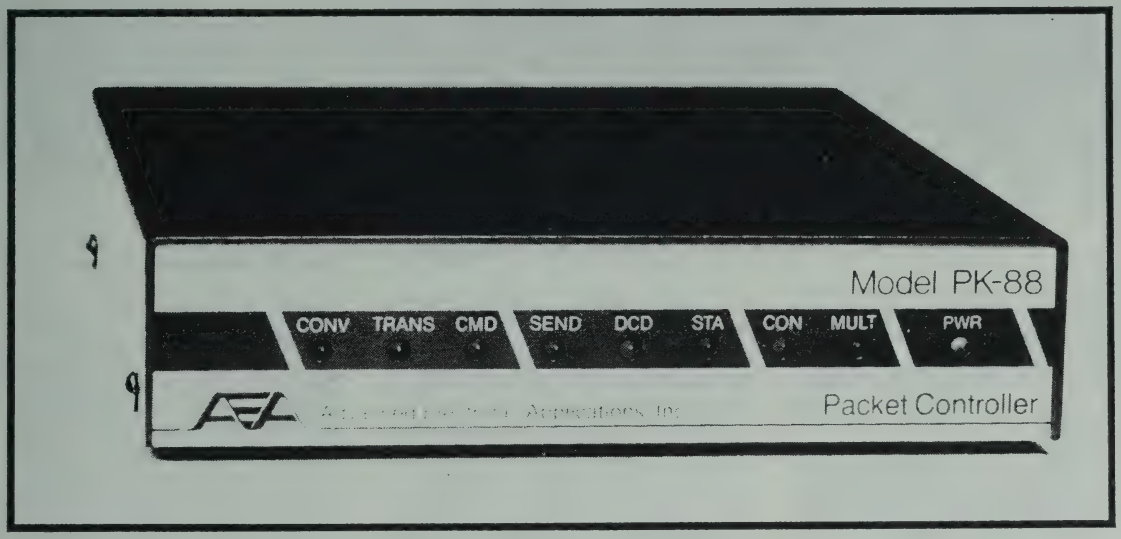
Enter the Host Mode. In this case, you bypass all the routines written to make the TNC compatible with the Human Interface. It is true that your computer is talking directly to the TNC's computer,

(cont'd on page 18)

New AEA PK-88

Packet Radio's Fourth Generation

AEA is proud of its contribution to the packet revolution in amateur radio. We were the first manufacturer to provide a commercial packet controller. Our newest controller, the PK- 88, is the fourth generation in a proud line.



Improved hardware and software design make the PK-88 your best choice for a packet only controller. Integrating the popular packet software from the multi-mode PK-232 with a special AEA TNC hardware design gives you the best of both worlds.

The new Maildrop feature of the PK-88 gives you the convenience of a personal mailbox. You can program 8K of memory in up to 15 different files with information that others can download. This information can be programmed for retrieval by a specific callsign, or made available to all connecting stations. Maildrop will also store incoming data sent to your station. The PK-88 is compatible with the popular TCP/IP protocol, and can be modified for NET ROM operation. The unique host mode of the PK-88 also gives programmers the ability to write special terminal programs (like AEA PC Pakratt and Com Pakratt programs) for full featured TNC use. Other software features of the PK-88 allow the operator to restrict the use of the station for both connects and as a digipeater. The mailbox monitoring command allows monitoring without displaying the callsign headers, while standard monitoring includes both MFROM and MTO lists.

For base station, portable or digipeater operation the PK-88 makes packet radio easy and affordable.

Amateur Net Price $119.95

AEA Brings you the Breakthrough

2006 - 196th St. SW • Lynnwood, WA 98036 • (206) 775-7373

by Jonathan L. Mayo, KR3T
3908 Short Hill Drive • Allentown, PA 18104

★★★ Portable RTTY Operation ★★★

Operating RTTY is one of my favorite Amateur Radio activities. I often leave my Rtty system running while I am away from the shack so that I can keep track of activity. Whenever I was away for an extended period of time, I often wished that I could take the RTTY station with me, but the physical size of the equipment along with its electrical requirements prevented me from doing so. This desire led to the development of the portable RTTY station described in this article.

A portable RTTY station has many advantages. Imagine being able to check into a RTTY net or mailbox while sitting in your car, motel room, or vacation cabin. Field Day RTTY operation would be very easy, and emergency traffic handling can utilize the advantages of RTTY while portable.

Theory

Any RTTY station must have three basic components; a terminal, a Terminal Unit (TU), and a transceiver. The terminal converts characters into digital codes and vice versa. The Terminal Unit converts digital signals to analog tones and vice versa. And finally, the transceiver transmits and receives the analog tones.

When transmitting, characters typed on the keyboard of a terminal are converted into digital codes which are sent to the Terminal Unit. In the Terminal Unit, the digital codes are converted to analog signals which are sent to the transmitter. The transmitter then transmits the analog signals.

At the receiving end, the receiver receives the analog signals and sends them to the Terminal Unit where they are converted back to digital codes. The digital codes are then sent to the terminal where they are converted back into characters which are sent to a display, printer, or other device. Either way, digital codes, analog signals, and conversions are involved.

Equipment

The equipment to be used in a portable environment must be carefully chosen to meet certain criteria. The equipment I use is listed below along with the reasons why it is suitable for a portable environment.

The TRS-80 Model 100, or any similar portable computer, serves as an ideal terminal. It is lightweight, compact, and battery operated. Its LCD screen is easy to read and does not draw much power. A built-in RS-232C port allows for easy connection to the Terminal Unit, and a built in terminal emulation program complete with upload and download capabilities makes operating easy.

Terminal Unit

Any Terminal Unit capable of battery operation will do. However, since I plan to operate portable only on the VHF bands, I use the Flesher TU1200. It is designed specifically for the VHF/UHF bands and can run at up to a 1200 baud data transmission rate.

Transceiver

Once again, any transceiver will do. For VHF work, a handie talkie such as the Icom 02-AT provides extreme portability.

All three pieces of equipment are very compact and are capable of battery operation. Thus, they are truly portable. Connecting the equipment together requires special cables which are discussed next.

Interfacing

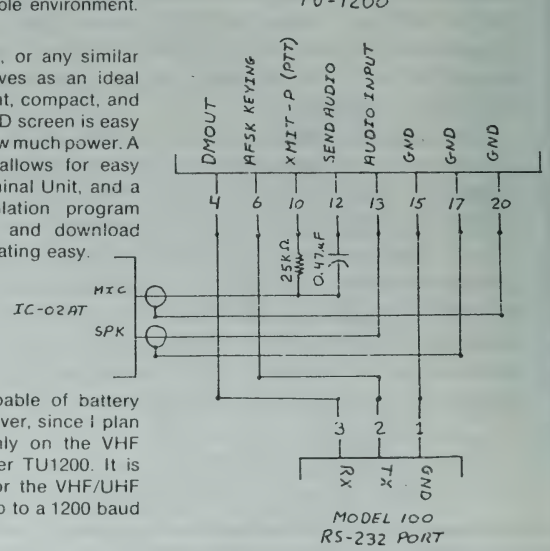
Since the interfacing for different equipment can vary, the wiring in *figure 1* only applies to my particular installation. In addition to the basic connections, I included a circuit to trip the PTT of the 02-AT when the SEND button is pushed on the TU1200.

Because the Model 100 uses RS232 level signals, it must be connected to compatible equipment. The TU1200 provides options for connections to both RS-232 and TTL level signals, so connecting the two was easy. Only three wires are needed; TX, RX, and GND.

The audio output of the 02AT goes straight to the TU. The 02-AT Mic input connects to the TU through a 0.47 uF capacitor. The capacitor is used to isolate the 02-AT PTT voltage from pin 12 of the TU, and the 25K ohm resistor is used to trip the radio's PTT circuit.

Now that both the terminal and the transceiver are connected to the TU, operation is possible.

(cont'd on page 15)



Join the fun at the top of the pack!

Whatever your interest, you'll find the **All Mode**™ from Kantronics at the top of the pack. CW, Radioteletype, AMTOR, ASCII, WEFAX and Packet, "**KAM**™ handles them all." HF and VHF simultaneous operation? You bet! The **KAM** can operate RTTY or AMTOR on HF while communicating Packet on VHF. Believe it. *It's the only true dual-port on the market.*

World-Class

Swedish and German hams have noticed **KAM**'s engineering and are some of our biggest fans. The Japanese scruitinized our features and gave us ideas for extras like a subject field for the **Personal Packet Mailbox**™, listings of digipeater paths and more CW characters and functions. We are shipping **KAM** around the world.

Fun for Everyone

You'll be the envy of the airwaves with all the modes and features **KAM** brings to your fingertips. In addition to Packet, ASCII, AMTOR, WEFAX, CW and Radioteletype, you'll have Dual-Port Operation (the only TNC offering this feature), **Personal Packet Mailbox**™, **KA-NODE**™ operation, **Gateway**™ operation and digipeater operation.

You can operate non-packet modes on HF while maintaining a packet station at the same time on VHF. Or run an HF/VHF accessible personal mailbox or pc-based BBS totally dedicated to packet. Configure your **KAM** as an HF to VHF digipeater (**Gateway**) or **KA-NODE**. Or open your LAN to the world. All these modes and more features are waiting to be discovered. Increase your station abilities to the most advanced available.

Today and Tomorrow

KAM is packed with all the latest features of today with Kantronics-written firmware. And since we write and copyright our own firmware, we're in a position to lead you into the exciting modes of tomorrow first. We have a history of keeping our customers current. Proof is in our firmware updates (2.82 in 1987 and 2.84 in 1988) offered at a fraction of the cost of a new purchase.

So join the fun, join the Top of the Pack.

Kantronics
RF Data Communications Specialists
1202 E. 23 Street Lawrence, Kansas 66046 (913) 842-7745

by Jonathan L. Mayo, KR3T
3908 Short Hill Drive • Allentown, PA 18104

Terminal Emulation Programs

— PART 1 —

In the last issue of Digital Digest, I gave an overview of microcomputer technology and applications in the amateur radio station. Amateurs have developed many innovative uses for microcomputers. Even as you read this sentence, microcomputers in amateur radio stations around the world are serving as terminals for digital communications, predicting the orbits of OSCAR satellites, designing optimal antenna systems, laying out printed circuit boards, and many other tasks.

Perhaps the most popular use of microcomputers in digital amateur radio stations is as terminals for packet radio TNCs (Terminal Node Controllers), RTTY TUs (RadioTeleType Terminal Units), or other digital communications interfaces. All microcomputers that act as digital communications terminals are running some sort of terminal emulation software, a program which allows a computer system to act like (emulate) a dedicated digital communications terminal. This article, the first of two, covers the basics of terminal emulator software. After reading this article, you should have a good understanding of what features you need in a terminal emulator software package for use in amateur radio digital communications.

Why Terminal Emulators?

The terminal emulator concept is very popular for several reasons. It is cost effective because there is no additional equipment to buy beyond the microcomputer system, and the microcomputer may be used for other purposes. Also, with a microcomputerbased terminal emulator, the information sent and received can be permanently saved to disk or cassette, formatted and printed using word processing software, or processed using database or spreadsheet software.

Terminal emulator software comes in a wide variety of capabilities. Some simply allow the computer to serve as a dumb terminal. Other more complex software allows the microcomputer to function as a sophisticated smart terminal. Some terminal emulator software is written specifically for packet or RTTY operation. This software is usually designed for use with a specific TNC or TU and contains features which enhance its use. Many TNC and TU manufacturers sell terminal emulation software written especially for their units. However, it is not necessary to have a specialized terminal program for radiobased digital communications.

Most ASCII (American Standard Code for Information Interchange) telecommunications software packages will work fine with radio-based digital communications units. These are the same programs used with telephonebased modems to access remote computer systems. There are a great number of terminal programs with an equally great diversity in features and capabilities. Generally, microcomputer systems with communications capabilities have terminal software available for them.

While terminal software can cost up to several hundred dollars, it is not necessary to necessary to spend vast sums of money. There are a great many programs available in the public domain that may be obtained at little or no cost. In addition, many fine shareware programs can be had for very reasonable prices. Often, operating system master disks include some sort of telecommunications capability, especially if the computer comes with a RS-232 port.

If you're not sure what kind of program is best or where to look for one for your particular computer, check with someone who is using the same type of computer you do. If you don't know of anyone, try to locate a user's group in your area; a good place to start is where you bought your computer. Be sure to look at a few back issues of computer magazines for ideas; check to see if there is one written specifically for your microcomputer system.

If you own a popular microcomputer system, such as an Apple, Commodore, IBM, or Tandy, you will have little trouble locating terminal software for your machine. In fact, you will probably be able to obtain specialized software written for your computer and a particular digital communications interface - I'll explore this further in Part II of this series.

Smart versus Dumb Terminals

I've used the words "smart" and "dumb" to describe terminal emulator software in the preceding section. Let's now discuss the differences between smart and dumb terminals.

A dumb terminal emulator has only the most basic of capabilities necessary for communications. It features very basic send and receive functions, most often just displaying received characters and transmitting exactly what is typed. Dumb terminals do not support expansion and do not offer much flexibility for modifying the communications characteristics. Most simple terminal programs can be classified as dumb; a good example of this are the simple telecommunications programs that sometimes come with microcomputer operating systems.

On the other hand, a smart terminal possesses advanced capabilities to enhance communications. Most sophisticated terminal emulation programs are classified as smart. There are many useful features available in smart terminals that are very nice to have when operating digital communications modes. The most useful features are explained in the next several paragraphs.

X-on and X-off capability is a very useful, almost mandatory feature for operating digital communications modes. In fact, some dumb terminals include this feature. The X-off signal, usually a CTRLS (Control S), tells the terminal to stop sending information. The X-on signal, usually a CTRL-Q, tells the terminal to resume sending the information.

This feature allows an attached device to automatically turn the flow of data from the terminal to the attached device on and off so that the device is not overloaded. For example, Xon/X-off is useful when a TNC is transmitting at a much lower baud rate (the speed at which the digital information is being sent between devices) than is being used between the terminal and the TNC. Assume the Tnc's modem is transmitting at 300 baud, and the terminal is sending the TNC data at 9,600 baud. The terminal is sending information to the TNC faster than the TNC can send it out, so the TNC buffers the information build-up in its memory. When the memory is almost full, the TNC sends the terminal an X-off signal to stop the flow of data. When the buffer space is again at an acceptable level, the TNC sends an X-on character to resume the flow of data.

Other control codes besides X-on/X-off are utilized heavily in digital communications to control the receiving terminal. Control codes such as clear screen, bell, and linefeed are commonly used in digital communications. A smart terminal program should allow for changing or filtering of control codes. For example, while many control codes are standardzed (ie. Linefeed, CTRL-J; Bell, CTRL-G), many are not. One terminal's clear screen control code might be another's delete transmit buffer code. One of my terminal programs uses CTRL-Z as the clear screen control code. Many text files use CTRL-Z as an EOF (End Of File)

(cont'd on page 15)

Computers *(cont'd from page 14)*

code, and whenever I received these files, my screen would go blank. By filtering out the CTRL-Z code before it was processed by the program, the screen would not clear at the end of a file.

The capability to download a file is extremely useful in all digital communications because it gives you the ability to save incoming data for later viewing or processing. On a microcomputer system, the data may be put in volatile memory and then saved to disk or cassette.

Terminal programs differ on downloading procedure, but most allow the user to open a buffer with a special keyboard sequence and close the buffer with another sequence after the desired data has been received. The buffer may then be cleared, displayed, printed, or saved on disk or cassette. These functions may be handled through special commands or menus.

Uploading is another useful feature that smart terminal emulators offer. Uploading allows for the sending of prepared data. The data may be saved on disk or in memory. In most systems, you specify which file or buffer is to be sent, and the data is automatically sent. This feature saves time and allows the same information to be sent to multiple destinations with a minimum of effort.

Keep in mind that many terminal emulator software packages have more features than I have listed here; these are the basics that you should look for in a smart terminal emulator for use with digital communications modes. The dividing line between smart and dumb terminals is not black and white, so attention should be given to the package's features rather than its designation. Some programs are smarter than others, but with increased capability usually comes increased complexity. The bottom line is to find a terminal that has the features you need and is comfortable for you to use.

Conclusion

This article introduced the subject of terminal emulation software for microcomputers. Using the information in this article, you should be able to weigh the features of various programs for your particular microcomputer system and select one that best fits your needs. In the next issue of Digital Digest, I'll continue the discussion of terminal emulators with a look at some popular terminal emulation software packages for various microcomputer systems.

Portable RTTY Operation *(cont'd from page 12)*

Operation

The Model 100's upload capabilities can be utilized to transmit pre-written CQs, "brag tapes", and other messages. Since its memory is non-volatile, the messages are not erased when the computer is turned off.

When receiving, the download capabilities can be used to maintain a record of what was received. A built in parallel printer port can be used to obtain a hardcopy on almost any printer.

Pushing the SEND button on the TU1200 causes the PTT on the 02-AT to trip, and any characters typed on the keyboard or uploaded from memory will be transmitted. Almost anything can be transmitted to another station, including messages, programs, and other data.

I have found that it is a good idea to carry an external antenna for use in situations where the rubber duckie is not adequate; such as in buildings. A simple, compact VHF "J" antenna can be fabricated out of ordinary twin-lead and hung on a curtain or from the ceiling. When not in use, the antenna can be rolled up and stored most anywhere. Other possibilities include the vertical dipole, ground plane, and quarter-wave whip.

When not operating RTTY, the handie talkie can be used for standard FM phone, and the Model 100 can be used as a standalone computer system.

While the Model 100's terminal program is designed for ASCII operation only, a program in either BASIC or machine code could be written to allow for Baudot operation. A TNC (Terminal Node Controller) could be plugged into the RS232 port for use as a Packet radio station. The possibilities are numerous and a portable station for most any mode of digital communications is possible.

Conclusion

RTTY has certainly changed over the past several years. Equipment which used to consume large amounts of space and require frequent maintenance has been shrunk to where it will easily fit in a briefcase. And RTTY has gone from one of the most mechanical and noisiest modes to one of the most electronic and quietest.

In addition to the actual hardware of digital communications, many advances have also been made beyond standard Baudot in the methods of encoding the messages. Throughout the past decade, modes such as ASCII, Amtor, and Packet have evolved to meet our need for more advanced capabilities.

Portable operation adds a new dimension to RTTY. It is a lot of fun to be able to work other stations or check into a mailbox system while traveling with a complete station that easily fits in a briefcase. And nothing beats digital communications for transmitting large amounts of data.

I hope this article provides you with some ideas for setting up your own portable RTTY station. If you already have a RTTY station and some sort of portable terminal, it is an easy task to set up a portable station to meet your needs. And if you're not yet equipped for digital communications, I strongly urge you to do so. It is a lot of fun and allows you to experiment with the future of radio communications.

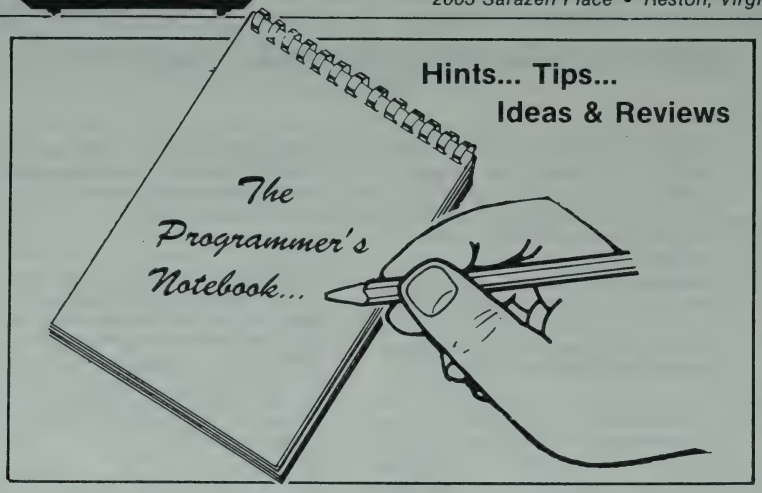
p-Persistence Table

#	0.50	0.25	0.10	0.01
2	24.3%	6.0%	1.0%	0.0%
3	49.6%	15.5%	2.8%	0.0%
4	68.5%	26.0%	5.3%	0.1%
5	81.5%	36.2%	8.6%	0.1%
6	89.5%	45.8%	11.3%	0.2%
7	93.4%	54.8%	14.9%	0.3%
8	96.2%	63.3%	19.0%	0.5%
9	97.9%	69.3%	23.0%	0.5%
10	99.1%	76.8%	27.0%	0.6%

(Table referenced from page 5)

by ***Peter G. Smith, N4ZR***

2003 Sarazen Place • Reston, Virginia 22091



It's an exciting thing to be in on the birth of a new publication. There's the chance to help set the tone, the personality that it will bring to its audience. Tom Arvo had already collected well-known authorities in all the key areas of digital amateur radio communications for Digital Digest when we discussed the possibility of this column, and I said that I thought I could bring a programmer's perspective to many of the same subjects. Not that of a professional, versed in the arcane world of assembly language, but of an ordinary ham interested in using digital techniques to expand his enjoyment of the hobby, who learned to program in order to meet his own needs, and then went beyond.

In the coming months, I hope to provide a stimulating mix of things in this space, including programming ideas, reviews of software from the perspective of someone who has worked to meet user needs from the inside, and opinion on a variety of topics. I don't promise that I'll always be right, and certainly not that you'll necessarily agree with me, but I hope it'll at least be interesting.

So you have some idea where I'm coming from, a bit of background would be helpful. I've been a ham since 1954, an active DX'er and contester, currently active on all modes except SSTV. I wrote my first program in 1984, when my father-in-law gave me a TimexSinclair 1000 with a cantankerous membrane keyboard and 16 KB of plugin RAM that always came loose just as you finished programming but before you saved it. Saving data to cassettes was another exciting aspect of working with that machine, and we can all be glad those days are gone.

Anyway, I used that first machine to write a simple dupechecking program in Sinclair semibasic, and I was hooked. I can remember wrestling with that program line-by-line, straining my brain to understand the logic of basic commands, learning about things like arrays (the Sinclair's instruction book was a decent place to start, at least).

About that time, my office got some Tandy TRS-80 Model 100 computers, mainly for use as traveling computer terminals for electronic mail. They had a good subset for MicroSoft Basic built in, and you couldn't be on electronic mail all the time you were on the road. Over the course of several months when I spent a lot of time in motel rooms, I wrote a little program for my own use that combined logging, duping, and sending CW. I don't even remember how I got the idea of marketing that program, but I did, and that's how Winner's Edge software and The Contester series of programs was born.

I've been at it now since mid-1985, and I'd guess my compensation over time has probably averaged about a penny an hour, but that's not the point. More than anything, I've enjoyed relationships with friends who became customers, and customers who became friends. I've enjoyed learning about the capabilities and limitations of particular computers and programming languages, and the intellectual stimulation of solving programming problems. Even if I'd never sold a piece of software, those payoffs would still be there.

My shack now demonstrates my involvement in this aspect of our hobby. The operating position is flanked by monitors for my Commodore 64 and IBM clone. The keyboard in use at the moment sits in front of me, and the radio is to one side.

WHY PROGRAM

As I've already suggested, I'm very glad I learned to program. I've talked about the intellectual challenges and satisfaction, but there's more to it than that. We live in an increasingly automated world. Almost all our monthly bills (and a lot of junk mail) come from computers. More and more appliances have embedded microprocessors. We have come to take computers for granted. But all too often, people of my mid-forties age group throw up their hands. "Too complicated," they say. "That stuff's for specialists, "they say.

Nuts. And double nuts. That's a mistake, for one obvious reason and one that, maybe, is not so obvious. The obvious one is that promotions, and better jobs, and more interesting assignments, will increasingly go to those who show their interest in, acceptance of, and knowledge of computers. What better way to flex those muscles than by learning how computers really work, through programming?

There's an insidious aspect to all this computerization, though, one which programming can also help you to handle. By now, we've probably all had experience with computerization or office automation projects that failed miserably. Usually, this is because they were oversold in the first place, by people who expected to solve their problems just by putting them on a machine. In fact, by going that route the main thing they did was to make their mistakes just that much more expensive.

So what does this have to do with programming? Simple... when you try to program a computer to do a task, however simple, it quickly becomes apparent that unless you understand the task you can't tell the computer how to do it. Programming forces you to address the logical structure of the job you're programming the computer to do. Successful programming also takes into account the needs and limitations of the human being on the other side of the keyboard, something that the designers of massive office automation projects forget all too often. Believe me, it's much harder to make that kind of mistake all blithe and unaware when you've written a few programs and wrestled with making them work right in the real world.

That's all the time and space I have for this issue - next time, I'll tell you why basic may be the only programming language you'll ever need, and review K1EA's real-time contest software for the IBM PC, the consensus choice of top contesters (and it's shareware!).

Stay Tuned!

Packet *(con'd from page 6)*

While ZIP Code traffic routing is still a bit controversial, the majority of PBBSs in the country have already established routing and forwarding tables to handle automatic message forwarding by ZIP codes. Until the packet community develops some "better" routing methods, ZIP code message routing **should** be used wherever possible.

When using ZIP Code routing addresses, you MUST use the normal five-digit ZIP Code. Partial ZIP Codes or so-called "wildcard" ZIP Codes (such as "11***") WILL NOT WORK!

There are several correct ways to SEND NTS messages:

a) If the addressee is a licensed amateur operator, with packet capabilities:
ST <callsign> @ <callsign>
where @ <callsign>" is the home BBS of the addressee

EXAMPLE:

To send an NTS message to "WX1AAA" at the "W2JUP" PBBS:
ST WX1AAA @ W2JUP
(Do NOT use the SSID (dash number) of the target BBS)

b) If the addressee is an amateur operator who is NOT on packet:

ST <callsign> @ <ZIPcode>
where "@ <ZIPcode> is the Postal ZIP code of the addressee or,

ST <callsign> @ <NTSXX>
where "XX" is the Postal Service symbol for destination STATE

EXAMPLE:

To send an NTS message to "WX1AAA" who lives in ZIP Code 11738:
ST WX1AAA @ 11738

EXAMPLE:

To send an NTS message to "WX1AAA" who lives in New York State:
ST WX1AAA @ NTSNY

EXAMPLE:

To send an NTS message to "WX1AAA" who lives in the New York City/ Long Island Section ("NLI"):
ST WX1AAA @ NTSNLI

c) If the addressee is NOT an amateur operator:
ST <ZIPcode> @ <NTSXX>

where "@ <ZIPcode> is the Postal ZIP code of the addressee
where "XX" is the Postal Service symbol for destination STATE

EXAMPLE:

To send an NTS message to a non-ham who lives in the Seattle area:

ST 98036 @ NTSWA

3. "SUBJECT" Field Clarifies Everything

Don't assume that each and every one of the more-than1000 PBBSs in the country already has its forwarding and routing tables set up to handle the specific PBBS to which your traffic needs to go. This assumption is false in 50% of all NTS traffic handled. Packet BBSs come and go with some regularity. Most of the PBBSs in the message-forwarding networks, with a few exceptions, are owned and operated by a private individual and are "on line" solely at that owner's discretion.

Most PBBS software provides a "SUBJECT" as part of each message header. This "SUBJECT" information also appears as part of the displayed information when users LIST the messages.

Rather than **waste** the "SUBJECT" field of the message on some relatively meaningless comment like "Hello from Disneyland", **use** the "SUBJECT" field to convey USEFUL routing information to the SysOps along the network. The information that you cleverly insert in the "SUBJECT" field may be all that a SysOp needs to route the traffic on its way. This information can help reduce the burden of service messages back along the network to the originator asking where the traffic is supposed to go.

In the following example of a "good" SUBJECT line, the telephone Area Code and Central Office Prefix are added after the name of the CITY and the standard abbreviation for the STATE.

The inclusion of the telephone Area Code and Central Office Prefix allows NTS Packet Liaison stations and any other packet operators to easily see if the destination of the traffic is within their Local Calling area, or if delivery by telephone would imply an interzone or toll call.

Note: Don't bother to put "QTC" in the SUBJECT line. Filing the message with the "ST" command already identifies the message as NTS radiogram traffic; that's exactly what the "T" means. "QTC" does mean "I have telegrams for.."; that's redundant with "ST".

In many cases studied over the years, the Area Code, telephone number and ZIP Code are frequently part of the original NTS Message Address, or may be available directly from the originator of the message. This is easy to obtain when filing messages from public events. Operators are urged to request this information from the people filing the traffic at the origin.

(cont'd on page 18)

Aries-1 *(cont'd from page 3)*

no longer on the list. The new version supports realtime printing and has allowances for printing out a short-form contest type log. Also, in version 1.3 is the ability to export log data files to dBase. Remaining shortcomings are the absence of a QSO buffer (a future version will contain one), and the inability of the contest logger to track multipliers or keep score.

It would be unfair not to mention the fact that ARIES1 also supports interfacing to various radios (late model Kenwood and ICOM units). Interfacing to these units would import frequency and mode automatically to the log. Ashton is currently developing further upgrades to add memories and more control of the transceiver through the interface. Also included is a log management and search module, and the ability to load external programs while ARIES stays RAM resident.

The software is available from Ashton, Inc., P.O. Box 1067, Vestal, NY 13851 for $89.95. It is compatible with the usual list of PC clones with a PK-232 or KAM interfaced to it. Additional TU's are not supsupported at this time. My setup is a Leading Edge model D2 (AT) and a PK232 with ICOM-751A which is not interfaced to the software.

Ashton's president is Thom Ashton (NY2I). I have spoken to him on the phone a few times and have given him my input on his software. He is very interested in user comments, and immediately incorporated some of my suggestions. Updates within 90 days of purchase are forwarded free of charge, and later updates are available at reduced rates.

In closing, let me say that ARIES-1 lives up to its claims 100 percent. It is well designed for contest operation and continues to receive support from its author. A pleasure to use, and easy to operate... it belongs in your software library.

Review by Tom Guntzel, KE0KB
-Source: RTTY Journal-

***** *Please Note* *****

If you have news and information of interest to the digital amateur community, please let us know!

— Editor —

EXAMPLE: 4. THE SAMPLE MESSAGE *(cont'd from page 17)*

The following information was compiled from ARRL form CD 218 2/83 by the NLI Section Traffic Manager, Bob Smutny K2MT. Questions on traffic handling may be sent to K2MT @ AI2Q-4. Your questions and involvment in traffic handling are welcome.

SAMPLE MESSAGE

```
 (1)(a) (b)  (c)   (d)       (e)          (f)   (g)      (h)
  1   R  HXE  K2MT  11  Valley Stream, NY   2100Z    Nov 21

(2) Joe Anybody
123 Any Avenue
Hamville, NY  12345
(123)-456-7890

(3) Hello Joe how are you query* will write later x** love

(4) Jane

(5)  (a) Rcvd K2MT 11/21/87 2200Z (b)Dlvd on phone 11/21/87 2300Z
     *The word QUERY is used in place of a question mark (?).
     **The letter X,(INITIAL X-RAY) is used in place of all other puntuation.

(1)(b) Precedence
    (EMERGENCY),(Always spelled out) Any message having life
       or death urgency to any person or group of people.
    (P),(Priority) Important messages having a specific time
       limit.
    (W),(Welfare) Either a query or advisory to the health
       and welfare of an individual in a disaster area.
    (R),(Routine) Most all traffic handled under normal
       circumstances.

(1) (c) Handling Instructions (Optional)
    HXA (Followed by number) Collect landline delivery
       authorized by addressee with in ..... miles.
    HXB (Followed by number) Cancel if not delivered
       within ..... hours of filing; service
       originating station.
    HXC Report date and time of delivery (TOD) to
       originating station.
    HXD Report to originating station the idenity of
       station from which received, plus date and time.
       Report idenity of station to which relayed, plus
       date and time, or if delivered report date, time
       and method of delivery.
    HXE Delivering station to get reply from addressee,
       originate message back.
    HXF (Followed by number) Hold delivery until .....
       (Date).
    HXG Delivery by mail or landline toll call not required.
       If toll or other expense involved, cancel message
       and service originating station.
```

A SERVICE MESSAGE is a message sent back to the station of origin if the message can't be delivered for any reason. Questions on traffic handling may be sent to K2MT @ AI2Q-4. Your questions and involvment in traffic handling are welcome.

Questions on any of these nets may be directed to the NLI Section Traffic Manager, K2MT @ AI2Q-4.

BITS & BYTES *(cont'd from page 10)*

but what you end up doing is re-writing all the code in the ROM to do more or less the same thing; interface with the operator. Why should you write a parser that is already written and in the system you are using. My approach is to build on the software provided by the TNC. If you need to test the TNC conditions, either use the Host Mode for that purpose only or use normal Language Parsing to test the responses from the TNC. An example is in the timing settings of a PK232. You can use host mode to tell your computer program what each setting is currently; or, you can simply send the settings you desire, which is exactly what you do after you find out what they already are. Why add the extra step? The better alternative is to write your own protocol and just use a modem. Try that!.

AMTOR *(Cont'd from page 8)*

selected is compatible with the tones your demodulator wants to see. This is where the Pass-Band-Tuning (PBT) or IF Shift (IFS) feature on a receiver is wonderful. Again, kudos to the designers at Drake and later Collins. The radio's from Asia now often come with IFS and that's a big help to the AMTOR operator. The PBT allows you to choose the frequency of the audio that your radio will output, with the chosen IF filter. Thus, for AMTOR, you set the PBT to provide frequencies in the 2100 to 2300 Hz range. When you want CW, you set it to 700 to 900 Hz and adjust the VFO slightly to place the signals in the IF passband. If you haven't experienced PBT find someone in your club who has a radio with this feature and ask him or her to demonstrate it to you. It's really slick for AMTOR!

Frequency control by synthesizer is wonderful for AMTOR but only if you have a rig that tunes in 10 Hz steps. You don't need to have the readout show the 10 Hz value (although that's nice) but it must tune in 10 Hz steps. Of course, VFOs provide continuous tuning so they don't have any problems, other than drift. If your radio jumps in 100 Hz steps, or more, you can sometimes be placed in a situation where you will be at least 50 Hz off frequency. With 170 Hz total shift this amounts to a 29% frequency error. Good modems won't work well, if at all, with that kind of frequency offset.

A feature that should almost come for free, yet many transceivers and receivers don't offer it, is a constant RX audio level output. That's an audio output with a constant level regardless of volume control setting. I mean really! The rigs have AGC so why not just grab the audio before the volume control, boost it up to 100 mVACrms and put it on the accessory jack? It's so simple to do and yet most radio manufacturers don't bother. When I set up a contact the last thing I want to do is to have to leave the volume up so the code converter has enough audio to decode the data. I want to turn the volume DOWN and not listen to the racket. sometimes I want to hear the audio quietly in the background, as a confidence check, but not usually. I know that some AMTOR enthusiasts build separate boxes to do this audio monitoring but why should one need to do this? Just use the radio's volume control to turn the audible volume up, down or off. What could be easier?

(con'd on page 19)

Computer Code... An Example

Each month I will try to give an example of some small sample of computer code that shows one way to accomplish a particular task. This month's example show's an example of how to test a callsign for validity using turbo prolog. I think that somewhere I have previously seen similar code used by WA7MBL for his BBS system. The comments should make the action self explanatory but to understand Prolog you will need to spend a little time reading. At the start of this routine the string is passed to the predicate test-call(string) which tests each letter of the string to determine whether it is a letter or numeral:

```
test_call(Call):-
        str_len(Call,Len),Len>=4,Len<=10,
        frontchar(Call,CH,Rest),                ltr(CH),
        frontchar(Rest,CH1,Rest1),              ltr(CH1),
        frontchar(Rest1,CH2,_),                 no (CH2),!
                or
        str_len(Call,Len),Len>=4,Len<=9,
        frontchar(Call,CH,Rest),                ltr(CH),
        frontchar(Rest,CH1,Rest1),              no (CH1),
        frontchar(Rest1,CH2,_),                 ltr(CH2),!
                or
        str_len(Call,Len),Len>=4,Len<=9,
        frontchar(Call,CH,Rest),                ltr(CH),
        frontchar(Rest,CH1,Rest1),              no (CH1),
        frontchar(Rest1,CH2,Rest2),             no (CH2),
        frontchar(Rest2,CH3,_),                 ltr(CH3),!
                or
        str_len(Call,Len),Len>=4,Len<=5,
        frontchar(Call,CH,Rest),                ltr(CH),
        frontchar(Rest,CH1,Rest1),              ltr(CH1),
        frontchar(Rest1,CH2,_),                 ltr(CH2),
        LLen=Len-1,
        frontstr(LLen,Call,_,SH3),
           str_char(SH3,CH3),
           CH3=('\42'),!.              /* ends in star */

        ltr(CH):-CH>=('\58'),CH<=('\90'),!. /* ltr */
        no (CH):-CH>=('\48'),CH<=('\57'),!. /* no */
```

The last predicates ltr() and no() simply are used by test—call() to find a number or letter. You may have picked out the three letters in the last test—call() routine. Note that it looks for three letters and an asterisk. This means that a packet DIGI that is transmitting with a TNC that identifies the transmitting station via the * can be recognized as legitimate. This kind of rountine can be as restrictive or permissive as the programmer desires. Just in case you are into Prolog, next month I'll give you an example of a much shorter routine that does the same thing using the string as a list of characters.

Lacy McCall //ac4x//

AMTOR *(cont'd from page 18)*

My last diddie, before I get off my soap box, is that rigs don't have balanced audio output (and the audio output should be constant level, regardless of volume control setting). Balanced simply means that neither side is referenced to ground. Usually the easiest way to accomplish this to pass the audio through a transformer. Because AMTOR uses narrow shift, the transformer doesn't need to be large and expensive. I use balanced audio around the shack because it provides a measure of RFI protection. I get balanced audio simply by passing it through cheap miniature transformers, like those from Radio Shack or Mouser. The dollar jobs are usually just fine. I like the 600:600 ohm units for unbalanced-to-balanced circuits. They are small and meet the need. Well, that's all for this issue. In a future article I hope to show you how you can add a balanced audio output, with constant level, to any transceiver or receiver. It's pretty simple and takes less than two hours to build the circuit, install it and test it. More on this in future articles. Next issue we will discuss the NAVTEX system of transmissions on 518 kHz and the NAVTEX protocol's application to ham radio. More info then.

If you received this issue as a complimentary copy, please fill out and mail the order form below to begin your subscription!

Get Connected With Digital Digest

Please fill out and mail with payment (U.S. Funds: Check or Money Order): ***$16.00*** *U.S. /* ***$20.00*** *Foreign*

Please check if NEW ☐ *or RENEWAL* ☐

Name _____ Call _____

Address _____

City _____ State ____ Zip ____

Favorite Modes _____

Special Interests _____

_____

Ham Station (Gear) _____

_____

Would you be interested in writing for Digital Digest? ☐ YES ☐ NO If yes, please state areas of interest _____

Please mail to: **Digital Digest,** 4063 N. Goldenrod Road, Winter Park, Florida 32792

Published by: ARVO & ASSOCIATES, INC. • Winter Park, FL

★ Publication & Subscription Information ★

Digital Digest is published six times per year, on or about the 15th of every other month commencing with the January/February issue.

★ **SUBSCRIPTION RATES** ★

$16.00 USA / $20.00 Elsewhere

Payable in U.S. Funds Only

Subscriptions should be mailed to:

Digital Digest
4063 N. Goldenrod Road,
Winter Park, Florida 32792

The purpose of the Digest is to provide useful and timely information on all modes and aspects of amateur digital communications.

Please submit letters, articles, club newsletters and other editorial material to: **Digital Digest**(Editorial Deptartment) 4063 N. Goldenrod Road • Winter Park Florida 32792

Digital Digest

Devoted entirely to Digital Radio Communications

4063 N. Goldenrod Road • Winter Park, Florida 32792

BULK RATE
U.S. POSTAGE
PAID
Permit No. 2449
Orlando, FL